First named inventor: McKenney   Page 9
Serial no. 10/813,470
Filed 3/30/2004
Attorney docket no. BEA920030022US1

REMARKS

Claim rejections under 35 USC 112

Claims 1, 3-6, 8-18, and 21-23 have been rejected under 35 USC 112, second paragraph, due to insufficient antecedent basis for the limitations "lock-free" and "look-ups." Applicant is respectfully confused about this rejection. Insofar as the independent claims 1, 8, 12, 15, and 21 recite both of these terms, reference to these terms in the dependent claims has proper antecedent basis. Insofar as both of these terms appear throughout the specification, reference to these terms in all the claims has proper antecedent basis. Therefore, Applicant requests the withdrawal of this rejection, or further explanation by the Examiner as to how, exactly, these claims lack antecedent basis as to the terms "lock-free" and "look-ups."

Claims 1, 3-6, 8-18, and 21-23 have also been rejected under 35 USC 112, second paragraph, as being indefinite. In particular, the Examiner has stated that the limitations "look-free" and "look-ups" have not been described in detail in the specification or the claims. Applicant respectfully disagrees. With respect to the term "look-ups," a look-up is simply a data access. This is stated in the patent application as filed, in which it is stated that, for instance, a "potential problem occurs when while a data file is being renamed or moved by a first process, a second process attempts to *access, or look up*, the data file." (P. 1, ll. 16-19) As such, the "second process may be able to *access* the data file, but mid-way through its *access* of the data file, the data file is successfully renamed or moved . . . such that the second process is no longer able to properly *access* the data file." (P. 1, ll. 19-23) Thus, this terminology "look up" is known within the art as simply being any type of data access, like a read or write for example. Applicant disagrees that there is anything indefinite about this terminology, since it is well known within the art.

First named inventor: McKenney
Serial no. 10/813,470
Filed 3/30/2004
Attorney docket no. BEA920030022US1

Page 10

With respect to the term "lock-free," in relation to a look-up, a lock-free look-up is simply a look-up, such as a data access, that does not utilize a lock. For example, the patent application as filed stated that "[a]tomicity can be provided for by the first process *locking* the file system and/or directories in which the data file is located, so that the second process cannot access the data file in any way until the first process completes its operations and *unlocks* the file system and/or directories." (P. 2, ll. 9-12) Thus, a lock-free look-up is simply that – a look-up in which a lock is not employed, since "locking requires significant overhead." (P. 2, l. 12) Applicant's entire specification is directed to how such lock-free look-ups are performed. For instance, FIGs. 2, 4, 5, and 6 described methods in which lock-free look-ups are permitted. In any case, Applicant disagrees that there is anything indefinite about this terminology, since locks are well known within the art. (For further information on what a lock is, Applicant points the Examiner to the Wikipedia entry for this term located on the Internet at http://en.wikipedia.org/wiki/Lock_%28computer_science%29 .)

Applicant notes that the case law on indefiniteness is as follows:

> In rejecting a claim under the second paragraph of 35 USC 112, it is incumbent on the examiner to establish that one of ordinary skill in the pertinent art, when reading the claims in light of the supporting specification, would not have been able to ascertain with a reasonable degree of precision and particularity the particular area set out and circumscribed by the claims.

(Ex parte Wu, 10 USPQ2d 2031, 2033 (BPAI 1989)) Here there is no lack of precision nor uncertainty, and thus no indefiniteness. Those of ordinary skill within the art readily understand what a "look-up" is. Likewise, those of ordinary skill within the art readily understand what a lock is, such that the terminology "lock-free" is also readily understood. There is nothing indefinite about the usage of either of these terms in the claims. Those of ordinary skill within the art know what they mean, and that is what it is important.

That said, Applicant's representative in particular can readily appreciate the Examiner's confusion as to the meaning of these terms to those of ordinary skill within the art. To that end, Applicant respectfully requests that she contact Applicant's representative, Mike Dryja, at the

First named inventor: McKenney   Page 11
Serial no. 10/813,470
Filed 3/30/2004
Attorney docket no. BEA920030022US1

phone number listed at the end of this response if further explanation is needed. For example, a teleconference can be set up with one of ordinary skill within the art to explain these terms further if such erudition is needed.

Claim rejections under 35 USC 101

Claims 1, 8, 12, 15, 18, 22, and 23 have been rejected under 35 USC 101 as being directed to non-statutory subject matter. The Examiner has made some suggestions as to various limitations that can be added to the claims to render them statutory. In speed along prosecution, Applicant has complied with the Examiner's request, and has added these suggestions to the claim language.

However, Applicant respectfully submits that the claimed invention most definitely is directed to statutory subject matter, in that it produces a "useful, concrete, and tangible result." The claimed invention generally permits lock-free lookups to data. This is incredibly useful, and is further concrete and tangible. In the background section of the patent application as filed, for instance, it is noted how computer systems can fail because a first process accesses data, and a second process acts on the data before the first process is finished accessing it, such that the first process fails. The background section further notes that locking can solve this problem, but that locking requires significant overhead, resulting in performance penalties. As such, lock-free lookups as to which the claimed invention is directed is useful – it allows for look-ups without suffering from performance penalties. Lock-free lookups are also concrete and tangible, in that there is really nothing more concrete or tangible in computer systems than the accessing of data. Insofar as the claimed invention allows for concurrent access to data without issues and without the performance problems as compared to employing locks, the invention is useful, concrete, and tangible.

Applicant provides a very rudimentary example in this respect to explain. Say your checking account balance is $1,000. You are depositing a check for $500 at an ATM, while at

First named inventor: McKenney   Page 12
Serial no. 10/813,470
Filed 3/30/2004
Attorney docket no. BEA920030022US1

the same time your bank is debiting your account by $250 due to an automatic bill payment. The process that updates your account balance starts first, and reads the checking account balance at $1,000. While it is adding $500 to your account, the process that debits your account balance then starts, and also reads the checking account balance at $1,000, since the $500 addition has not yet been completed. The former process then updates the total to $1,500, since it added $500 to the $1,000 original balance. The latter process then updates the total to $750, since it subtract $250 from the $1,000 original balance. This is bad news – your account now reads $750, when in fact it should read $1,000 - $250 + $500 = $1,250!

Therefore, one easy solution, as in the prior art, is to use locks. Thus, when the process that is adding $500 to your account starts, it "locks" the data holding your checking account balance. While it is adding the $500 to your account, the process that wants to debit your account tries to access this data to subtract $250, but it cannot obtain the data, because the former process has placed a lock on the data. Therefore, the latter process has to wait. Once the first process has added $500 to your account, to yield $1,500 total, and has unlocked the data in question, only then can the latter process access the account balance of $1,500 to subtract $250 therefrom to yield the correct $1,250.

However, as noted in the background section, "locking requires significant overhead, both on the first process that is doing the locking and unlocking, as well as on the second process that is looking up the data file." (P. 2, ll. 12-14) Lock-free lookups are thus a very real, concrete, and tangible issue in any type of data-processing system in which multiple processes may access data. Think of any multiple-user database, any stock market exchange, etc. All of these examples benefit from the claimed invention, insofar as the claimed invention provides for look-ups to data in which locks are not used, such that the performance penalties associated with locks are not present. The invention is most definitively useful, as well as concrete and tangible!

Claim rejections under 35 USC 102

First named inventor: McKenney   Page 13
Serial no. 10/813,470
Filed 3/30/2004
Attorney docket no. BEA920030022US1

Claims 1-23 have been rejected under 35 USC 102(e) as being anticipated by Stockley (6,865,583). Applicant respectfully traverses this rejection. That is, in light of the above added explanation as to what lock-free lookups are, such that the claimed invention permits them while renaming, moving, etc., a given file, it should be evident that Stockley does not disclose these limitations of the claimed invention.

As one example, the Examiner has shown how Stockley provides for the renaming of a data file in columns 5, 6, 7, and 8. However, the important element of the claimed invention is that a data file can be renamed while permitting lock-free look-ups of data. This Stockley does not disclose. Indeed, it is not even clear that Stockley is performed in relation to a multi-threaded, or multiple-process, architecture. That is, Stockley does not even disclose that while one process is renaming a file, another process is attempting to look up the data of the file. As such, Stockley is very much irrelevant to the claimed invention. If there is only a single process, then while a given file is being renamed by the process, of course no look-ups of the data can be achieved (insofar as the only process that is running is actively renaming the file), let alone lock-free look-ups as in the claimed invention. Furthermore, if Stockley were somehow made to operate in a multi-threaded or multiple-process environment, one of ordinary skill within the art could not add lock-free look-up capability, since such lock-free look-up capability is only taught in the claimed invention. Rather, one of ordinary skill within the art, having the prior art available to him or her, would add lock capability to Stockley so that it can operate in a multiple-process or multi-threaded environment.

In other words, the claimed invention in the various claims is limited to "doing [X]" . . . "while permitting lock-free look-ups." Stockley discloses "doing [X]," period. Stockley does not disclose "permitting lock-free look-ups." Insofar as Stockley is silent as to being operable in a multiple-process or multi-threaded environment, if you assume that it operates in a single-process environment, then at best Stockley cannot even permit look-ups while "doing [X]," since the only process that is currently running would be "doing [X]" such that look-ups could not be

First named inventor: McKenney   Page 14
Serial no. 10/813,470
Filed 3/30/2004
Attorney docket no. BEA920030022US1

concurrently accomplished.   Now, if you modify Stockley to operate in a multiple-process environment, then you have to look at how the prior art allows for solving the concurrency problem – and as stated in the patent application as filed, locking would be used, such that Stockley in this instance would be "doing [X] while permitting "look-ups *using locks*" and not "lock-free look-ups" as in the claimed invention.

For these reasons, therefore, Stockley does not anticipate the claimed invention.   As a parenthetical comment, Applicant submits that prior art of the type that Stockley represents is not really pertinent to the claimed invention.   Things like locking, concurrent access, multiple processes, etc., are usually found in lower-level computer science technologies, such as relating to file systems and operating systems. Most higher-level computer science applications rely on these lower-level technologies to handle these issues, and there is nothing in Stockley to suggest differently. The only exception would be databases, which normally and typically employ locking to solve concurrency problems.   Therefore, Applicant respectfully suggests that the Examiner look to lower-level computer science prior art in relation to the present patent application. However, Applicant has conducted a thorough prior art search, the results of which were provided in the Information Disclosure Statement (IDS) filed at the time of filing of the present patent application, and could locate no prior art that would suggest that the invention as recited in the claims is unpatentable.

First named inventor: McKenney   Page 15
Serial no. 10/813,470
Filed 3/30/2004
Attorney docket no. BEA920030022US1

Conclusion

Applicants have made a diligent effort to place the pending claims in condition for allowance, and request that they so be allowed. However, should there remain unresolved issues that require adverse action, it is respectfully requested that the Examiner telephone Mike Dryja, Applicants' Attorney, at 425-427-5094, so that such issues may be resolved as expeditiously as possible. For these reasons, this application is now considered to be in condition for allowance and such action is earnestly solicited.

Respectfully Submitted,

December 18, 2006
Date

Michael A. Dryja, Reg. No. 39,662
Attorney/Agent for Applicant(s)

Law Offices of Michael Dryja
704 228th Ave NE #694
Sammamish, WA 98074
tel: 425-427-5094
fax: 206-374-2819